

PAPER • OPEN ACCESS

Hydrodynamical Simulation of Astrophysical Flows: High-Performance GPU Implementation

To cite this article: Elena Akimova *et al* 2019 *J. Phys.: Conf. Ser.* **1336** 012014

View the [article online](#) for updates and enhancements.



IOP | ebooks™

Bringing together innovative digital publishing with leading authors from the global scientific community.

Start exploring the collection—download the first chapter of every title for free.

Hydrodynamical Simulation of Astrophysical Flows: High-Performance GPU Implementation

Elena Akimova^{a,b}, Vladimir Misilov^{a,b}, Igor Kulikov^c, Igor Chernykh^c

^aN. N. Krasovskii Institute of Mathematics and Mechanics UrB RAS, Ekaterinburg, Russia

^bUral Federal University, Ekaterinburg, Russia

^cInstitute of Computational Mathematics and Mathematical Geophysics SB RAS, Novosibirsk, Russia

E-mail: v.e.misilov@urfu.ru

Abstract. We present a new hydrodynamical code GPUPEGAS 2.0 for 3D simulation of astrophysical flows using the GPUs. This code is an extension of GPUPEGAS code developed in 2014 for simulation of interacting galaxies. GPUPEGAS 2.0 is based on the Authors' numerical method of high order of accuracy for smooth solutions with small dissipation of the solution in discontinuities. The high order of accuracy and small dissipation are achieved by using the piecewise-linear representation of the physical variables in each dimension. The Rusanov flux allows one to simply vectorize the solution of the Riemann problem. The code was implemented for the cluster supercomputers NKS-30T (Siberian Supercomputer Center, SB RAS) and Uran (Institute of Mathematics and Mechanics, UrB RAS) using the hybrid MPI+CUDA technology. To avoid the compute capability-specific implementations of reduction routines, the Thrust library was used. The optimal parameters for kernel function were found for the three-dimensional computation grid. The Sedov point blast problem was used as a main test one. The numerical experiment was performed to simulate the hydrodynamics of the type II supernova explosion for the grid size of 256^3 . A set of experiments was performed to study performance and scalability of the developed code. The performance of 25 GFLOPS was achieved using a single Tesla M2090 GPU. The speedup of 3 times was achieved using a node with 4 GPUs. By using 16 GPUs, 70% scalability was achieved.

1. Introduction

Studying the hydrodynamics of the supernova explosions is an important astrophysical problem. Naturally, because of the proportion of scales between the stars, the supernova remnant, and the core, the simulation of such explosion requires the most powerful supercomputers. Most of the modern supercomputers use the hybrid architecture, namely, the graphics accelerators (GPUs) and multicore coprocessors. Development of codes for such architectures is a difficult problem. Previously, the Authors had implemented the GPUPEGAS code [1] for supercomputers with graphics processors and the AstroPhi code for Intel Xeon Phi in the offload mode and in native mode using the low-level vectorization [2].

In this work, we present a new code GPUPEGAS 2.0 for 3D simulation of astrophysical flows using the GPUs. The code is based on the efficient piecewise-parabolic method on the local stencil [3, 4], which was successfully applied to a number of astrophysical problems [5, 6]. The main advantages of this method is the simplicity and cost-efficiency while retaining the features of the high order of accuracy and small dissipativity.



2. Numerical Model

We will briefly describe the problems and the numerical methods. Let us consider the conservative form of the equations of the gravitational hydrodynamics in the three-dimensional Cartesian coordinates for the law of conservation of mass

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}) = 0, \quad (1)$$

for the law of conservation of momentum

$$\frac{\partial \rho \vec{u}}{\partial t} + \nabla \cdot (\rho \vec{u} \vec{u} + p) = -\rho \nabla \Phi, \quad (2)$$

for the law of conservation of total mechanical energy

$$\frac{\partial}{\partial t} \left[E + \rho \frac{\vec{u}^2}{2} \right] + \nabla \cdot \left(\left[E + p + \rho \frac{\vec{u}^2}{2} \right] \vec{u} \right) = -(\rho \nabla \Phi, \vec{u}) \quad (3)$$

complemented by the Poisson equation for the gravitational potential

$$\Delta \Phi = 4\pi G \rho, \quad (4)$$

where ρ is the density, \vec{u} is the velocity, p is the pressure, Φ is the gravitational potential, E is the intrinsic energy, G is the gravitational constant.

The initial conditions are drawn from the statement of the problem. In our code, we use the periodic boundary conditions. We will consider two sets of variables: the conservative variables $U = \left(\rho, \rho \vec{u}, E + \rho \frac{\vec{u}^2}{2} \right)^\top$ and physical (or primitive) variables $P = (\rho, \vec{u}, p)^\top$. The conservation laws (1, 2, 3) can be rewritten in the vector form

$$\frac{\partial U}{\partial t} + \nabla \cdot F(U) = Q, \quad (5)$$

where $F(U)$ is the flux vector of the conservative variables, Q is the right part vector describing the work of the gravitational forces. For discretizing the computational 3D domain, let us introduce regular (on each dimension) rectangular grid of $I \times K \times L$ dimension, where h_x , h_y , h_z are the grid steps.

The time step τ is calculated from the Courant condition

$$\frac{\tau \times (c + \|\vec{u}\|)}{h} = CFL < 1, \quad (6)$$

where CFL is the Courant number, $c = \sqrt{\gamma p / \rho}$ is the speed of sound.

The conservative and primitive variables are defined at the centers of cells. As a result, for the vector form of the conservation laws (5), let us write the Godunov type scheme

$$\begin{aligned} & \frac{U_{i+\frac{1}{2}, k+\frac{1}{2}, l+\frac{1}{2}}^{n+1} - U_{i+\frac{1}{2}, k+\frac{1}{2}, l+\frac{1}{2}}^n}{\tau} + \frac{F_{i+1, k+\frac{1}{2}, l+\frac{1}{2}}^x - F_{i, k+\frac{1}{2}, l+\frac{1}{2}}^x}{h} + \\ & \frac{F_{i+\frac{1}{2}, k+1, l+\frac{1}{2}}^y - F_{i+\frac{1}{2}, k, l+\frac{1}{2}}^y}{h} + \frac{F_{i+\frac{1}{2}, k+\frac{1}{2}, l+1}^z - F_{i+\frac{1}{2}, k+\frac{1}{2}, l}^z}{h} = Q_{i+\frac{1}{2}, k+\frac{1}{2}, l+\frac{1}{2}}^n. \end{aligned} \quad (7)$$

For implementing the scheme (7), we need to define the values of fluxes at the interfaces between the cells. To achieve this, we have constructed the piecewise-linear representation of the primitive variables P at each cell of the computational domain for each direction using the minmod limiter.

As a Riemann solver for the right (R) and left (L) cell, we use the Rusanov flux or the Lax–Friedrichs flux

$$\mathcal{R}(P^R, P^L) = \frac{F(P^L) + F(P^R)}{2} + \frac{\max(c(P^R) + \|\vec{u}^R\|, c(P^L) + \|\vec{u}^L\|)}{2} (U(P^L) - U(P^R)). \quad (8)$$

For reconstructing the gravitational potential using the density function, we use the 27 point template for approximating the Poisson equation. This arises from the need for maximum invariance of the solution with respect to rotation. The density function and the potential are written in form of the superposition of eigenfunctions of the Laplace operator. To perform this, the fast Fourier transform is used. After substitution of this superposition to the approximation scheme of the Poisson equation, we get the fairly simple expression for calculation the harmonic magnitudes of the potential

$$\phi_{jmn} = \frac{\frac{2}{3}\pi G h^2 \sigma_{jmn}}{1 - \left(1 - \frac{2 \sin^2 \frac{\pi j}{I}}{3}\right) \left(1 - \frac{2 \sin^2 \frac{\pi m}{K}}{3}\right) \left(1 - \frac{2 \sin^2 \frac{\pi n}{L}}{3}\right)}.$$

where I, K, L are the numbers of cells for each coordinate. After this, we need to perform the reverse fast Fourier transform of the potential magnitudes to the functional space of the harmonics.

3. Parallel Implementation

We will describe the architecture of the code and the specifics of implementation using the MPI and CUDA technologies. Developed numerical method for solving the hydrodynamic equations allows one to apply the geometrical domain decomposition using one layer for subdomains overlapping. To solve the Poisson equation on the distributed memory supercomputer on the basis of the fast Fourier transform, the FFTW library was used.

The GPUPEGAS 2.0 code uses the multilevel multidimensional decomposition of domain. The outer one-dimensional decomposition into set of subdomains is performed along one coordinate axes using MPI tools. The decomposition of each subdomain is performed using CUDA tools as shown in Fig 1.

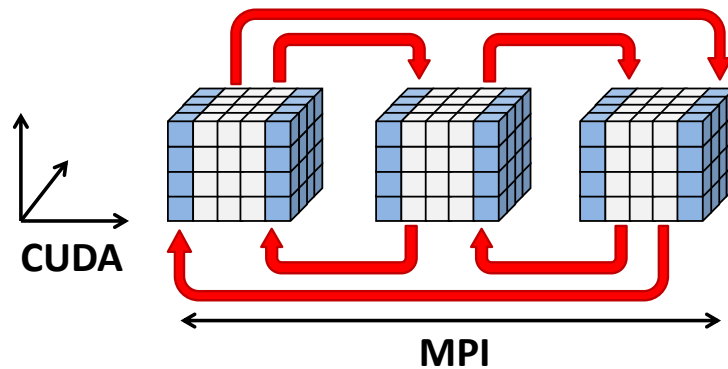


Figure 1. Geometric domain decomposition.

This structure of the computation provides a potentially infinite scalability.

The following algorithm for computing a single time step is used:

- (i) Find a minimum time step between all spatial cells (on GPU; the reduction is performed by means of the Thrust library).

- (ii) Solving the Poisson equation in a harmonic space (on CPU; the forward and inverse Fourier transforms are performed by means of the FFTW library).
- (iii) Linear reconstruction of the primitive variables (on GPU).
- (iv) Transferring reconstructed primitive variables in the overlapped subdomain boundaries between the MPI processes.
- (v) Solving the Riemann problem for each interface of each cell by the Riemann solver (on GPU).
- (vi) Calculating the values of the primitive variables (on GPU).

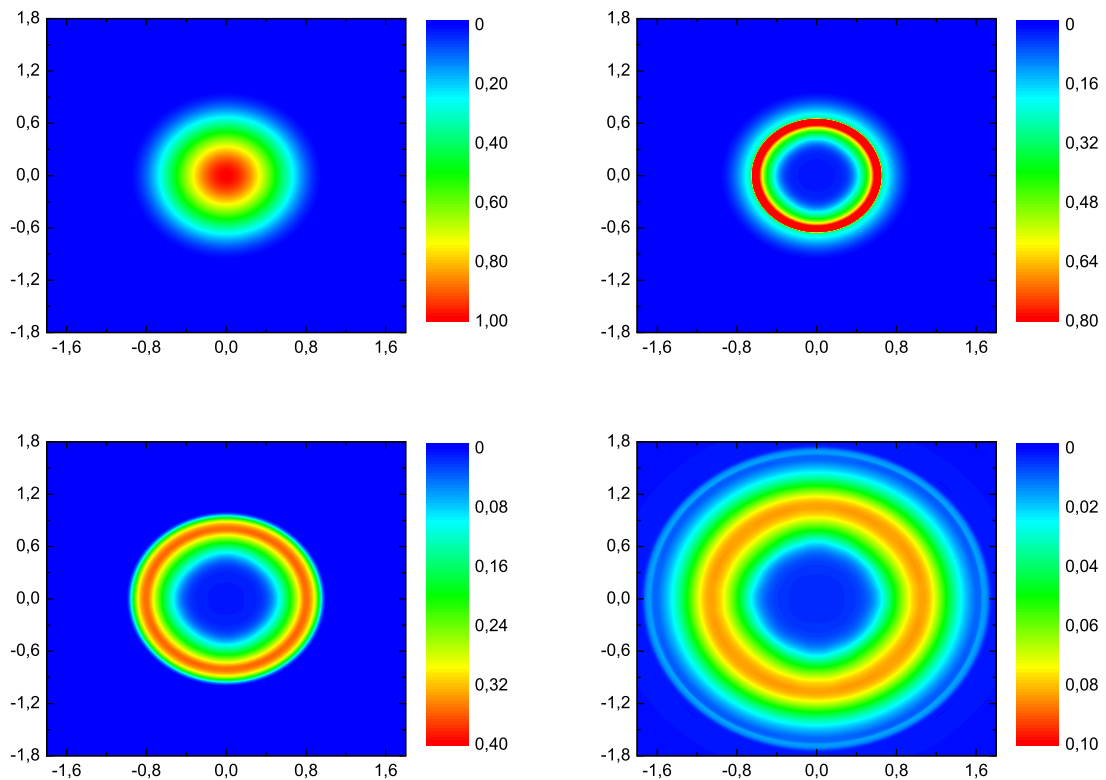


Figure 2. Density of the supernova in the equatorial cut of the star at the dimensionless instants of time 0 (top left), 0.1 (top right), 0.2 (bottom left), and 0.4 (bottom right).

4. Numerical experiments

As a test problem, we use the Sedov blast problem. For simulating the problem of point blast, we will consider the domain $[0; 1]^3$, the isentropic exponent is $\gamma = 5/3$, the initial density in the area is $\rho_0 = 1$, and the initial pressure is $p_0 = 10^{-5}$. At the instant $t = 0$, the intrinsic energy $E_0 = 0.6$ is emitted. The blast area is constrained by the radius $r_{\text{central}} = 0.02$. For the numerical experiments, the computational grid size of 256^3 was used. The simulated density profiles at the various instants t are shown in Fig. 2.

The Sedov blast problem is the standard test for checking the capability of the method and its implementation to reconstruct powerful shockwaves with large Mach numbers. The speed of sound in the background is extremely low; thus, the Mach number reaches the value of $M \approx 10^3$.

The Authors' numerical method fairly well reconstructs the location of the shockwave, as well as, the profiles for density and momentum.

The experiments were performed on the Tesla M2090 accelerators incorporated into the NKS-30T cluster of the Siberian Supercomputer Center. For studying the strong scalability, we use the speedup $S_m = T_1/T_m$ and efficiency $E_m = S_m/m$ coefficients, where T_m is calculation time on m devices for the same problem. Table 1 shows the results of solving the problem with the 256^3 mesh on various number of GPUs.

Table 1. Results of strong scaling experiments.

Number of GPUs m	Time per iteration T_m , msec	Speedup S_m	Efficiency E_m
1	1696	—	—
2	924	1.84	0.92
4	581	2.92	0.73

For studying the weak scalability, each of the GPUs stores and processes a portion of the whole mesh. The size of the portion is equal to 256^3 . Table 2 shows the efficiency of solving the problem with the fixed portion size per GPU.

Table 2. Results of weak scaling experiments.

Number of GPUs m	Time per iteration T_m , msec	Efficiency T_1/T_m
1	1696	—
2	1787	0.95
4	1862	0.91
8	2055	0.82
16	2420	0.70

5. Conclusion

We presented the novel hydrodynamic GPUPEGAS 2.0 code for the three-dimensional simulation of the astrophysical flows using graphics accelerators. Within one GPU we achieved the performance of 25 GFLOPS. By using single node with 4 GPUs we achieved the 3-fold speedup. By using 16 GPUs, we achieved the scalability of 70%. The numerical experiment was performed for the test Sedov point blast problem.

Acknowledgments

The work of Igor Kulikov and Igor Chernykh was supported by Russian Science Foundation (project no. 18-11-00044).

References

- [1] Kulikov I 2014 *Astrophys. J. Suppl. Ser.* **214**(1) 12
- [2] Kulikov I M, Chernykh I G, Glinskiy B M, Protasov V A 2018 *Lobachevskii J. Math.* **39**(4) 543–50
- [3] Popov M, Ustyugov S 2007 *Comp. Math. & Math. Phys.* **47**(12) 1970–89
- [4] Popov M, Ustyugov S 2008 *Comp. Math. & Math. Phys.* **48**(3) 477–99
- [5] Kulikov I, Vorobyov E 2016 *J. Comp. Phys.* **317** 318–46
- [6] Kulikov I M, Chernykh I G, Tutukov A V 2018 *Lobachevskii J. Math.* **39**(9) 1207–16